

School of Computer Science
UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
DEHRADUN, UTTARAKHAND



**System Monitoring and Configuration
Management**

Lab File

(2024)

for

6th Semester

Submitted To:

Dr. Hitesh Kumar Sharma

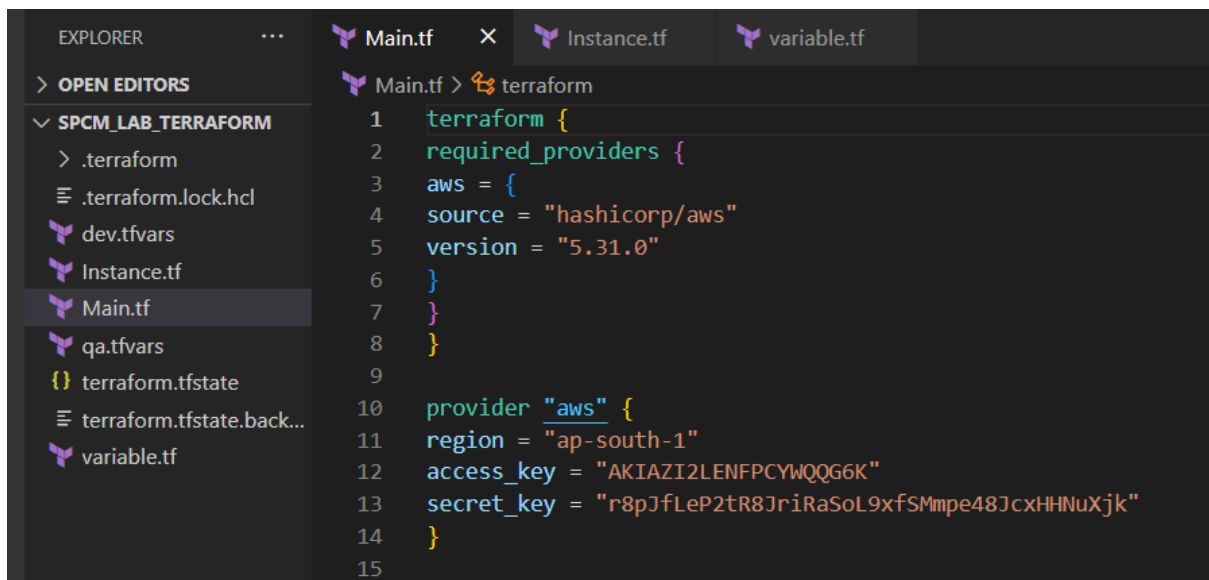
Submitted By:

Eksha Malhotra
B. Tech. CSE DevOps
[6^h Semester]
Sap id- 500091532
Batch 1
R2142210300

LAB EXERCISE 7

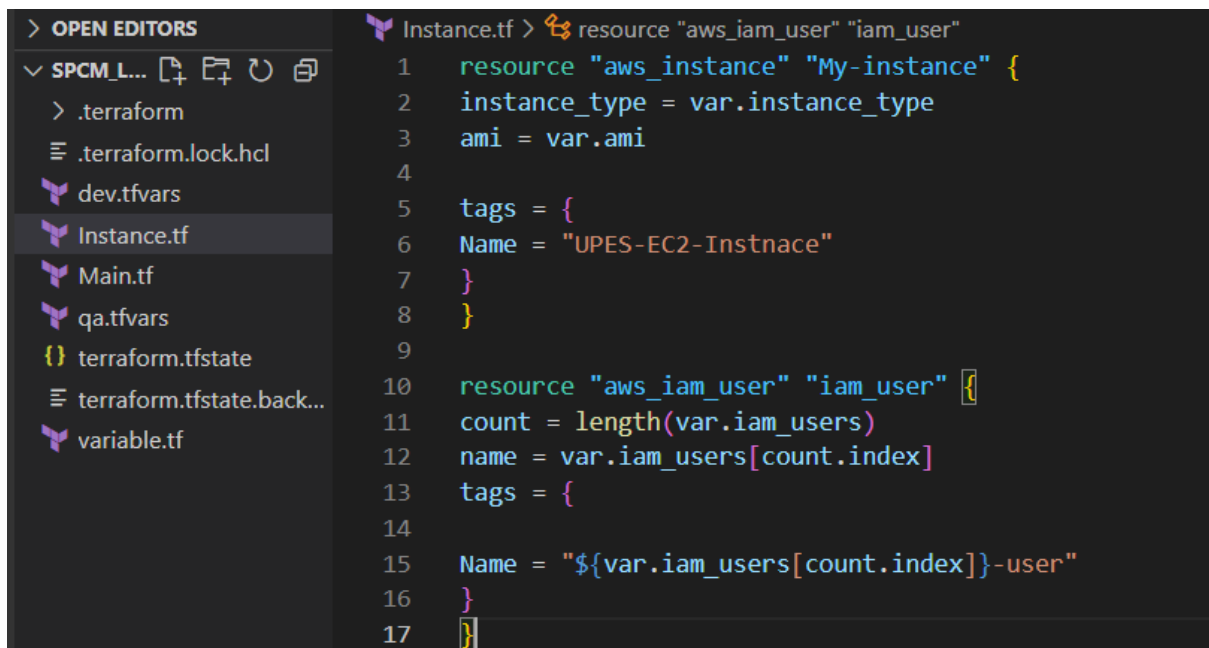
Aim: Creating Multiple IAM Users in Terraform

Step 1: Create a main.tf file.

A screenshot of the Visual Studio Code editor interface. The Explorer sidebar on the left shows a project named 'SPCM_LAB_TERRAFORM' with files including '.terraform', '.terraform.lock.hcl', 'dev.tfvars', 'Instance.tf', 'Main.tf' (selected), 'qa.tfvars', 'terraform.tfstate', 'terraform.tfstate.back...', and 'variable.tf'. The Editor pane shows the 'Main.tf' file with the following Terraform configuration:

```
1 terraform {
2   required_providers {
3     aws = {
4       source = "hashicorp/aws"
5       version = "5.31.0"
6     }
7   }
8 }
9
10 provider "aws" {
11   region = "ap-south-1"
12   access_key = "AKIAZI2LENFPCYWQQG6K"
13   secret_key = "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"
14 }
15
```

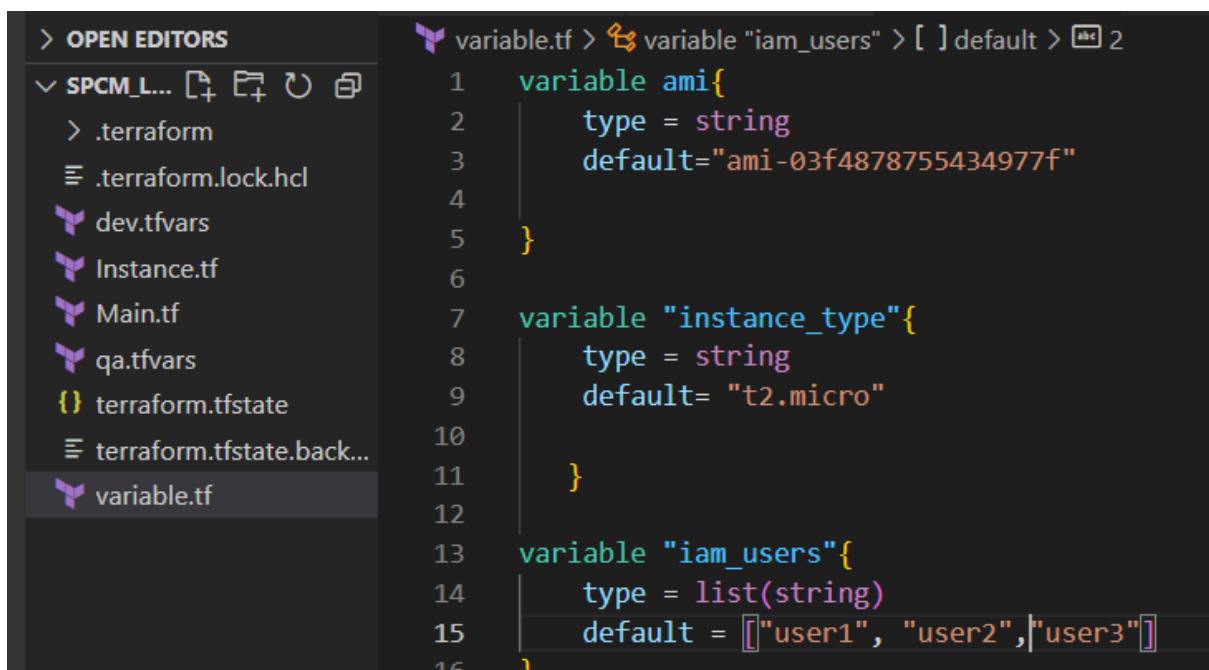
Step 2: Create a instance.tf file



The screenshot shows a code editor with a sidebar on the left titled "OPEN EDITORS". It lists several files: `SPCM_L...`, `.terraform`, `.terraform.lock.hcl`, `dev.tfvars`, `Instance.tf` (selected), `Main.tf`, `qa.tfvars`, `terraform.tfstate`, `terraform.tfstate.back...`, and `variable.tf`. The main editor area shows the content of `Instance.tf`. It contains two Terraform resource blocks. The first block defines an `aws_instance` resource named `My-instance` with attributes `instance_type = var.instance_type` and `ami = var.ami`. The second block defines an `aws_iam_user` resource named `iam_user` with attributes `count = length(var.iam_users)`, `name = var.iam_users[count.index]`, and `tags = { Name = "${var.iam_users[count.index]}-user" }`.

```
Instance.tf > resource "aws_iam_user" "iam_user"
1  resource "aws_instance" "My-instance" {
2    instance_type = var.instance_type
3    ami = var.ami
4
5    tags = {
6      Name = "UPES-EC2-Instnace"
7    }
8  }
9
10 resource "aws_iam_user" "iam_user" {
11   count = length(var.iam_users)
12   name = var.iam_users[count.index]
13   tags = {
14
15     Name = "${var.iam_users[count.index]}-user"
16   }
17 }
```

Step 3: Create a variable.tf file



The screenshot shows a code editor with a sidebar on the left titled "OPEN EDITORS". It lists several files: `SPCM_L...`, `.terraform`, `.terraform.lock.hcl`, `dev.tfvars`, `Instance.tf`, `Main.tf`, `qa.tfvars`, `terraform.tfstate`, `terraform.tfstate.back...`, and `variable.tf` (selected). The main editor area shows the content of `variable.tf`. It contains three Terraform variable blocks. The first block defines a variable `ami` of type `string` with a default value of `"ami-03f4878755434977f"`. The second block defines a variable `instance_type` of type `string` with a default value of `"t2.micro"`. The third block defines a variable `iam_users` of type `list(string)` with a default value of `["user1", "user2", "user3"]`.

```
variable.tf > variable "iam_users" > [ ] default > abc 2
1  variable ami{
2    type = string
3    default="ami-03f4878755434977f"
4  }
5
6
7  variable "instance_type"{
8    type = string
9    default= "t2.micro"
10
11  }
12
13 variable "iam_users"{
14   type = list(string)
15   default = ["user1", "user2", "user3"]
16 }
```

Step 4: Now initializes

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform init
```

Initializing the backend...

Initializing provider plugins...

- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.31.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see any changes that are required for your infrastructure. All Terraform commands should now work.

If you ever set or change modules or backend configuration for Terraform, rerun this command to reinitialize your working directory. If you forget, other commands will detect it and remind you to do so if necessary.

Step 5: Now perform validate

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform validate
Success! The configuration is valid.
```

Step 6: Now perform the terraform apply

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform validate
Success! The configuration is valid.

F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
+ create

Terraform will perform the following actions:

# aws_iam_user.iam_user[0] will be created
+ resource "aws_iam_user" "iam_user" {
+   arn                = (known after apply)
+   force_destroy      = false
+   id                 = (known after apply)
+   name               = "user1"
+   path               = "/"
+   tags               = {
+     "Name" = "user1-user"
+   }
+   tags_all           = {
+     "Name" = "user1-user"
+   }
+   unique_id          = (known after apply)
}

# aws_iam_user.iam_user[1] will be created
+ resource "aws_iam_user" "iam_user" {
+   arn                = (known after apply)
+   force_destroy      = false
+   id                 = (known after apply)
+   name               = "user2"
+   path               = "/"
+   tags               = {
+     "Name" = "user2-user"
+   }
+   tags_all           = {
+     "Name" = "user2-user"
+   }
+   unique_id          = (known after apply)
}

# aws_iam_user.iam_user[2] will be created
+ resource "aws_iam_user" "iam_user" {
+   arn                = (known after apply)
+   force_destroy      = false
+   id                 = (known after apply)
+   name               = "user3"
+   path               = "/"
+   tags               = {
+     "Name" = "user3-user"
+   }
+   tags_all           = {
+     "Name" = "user3-user"
+   }
+   unique_id          = (known after apply)
}

# aws_instance.My-instance will be created
+ resource "aws_instance" "My-instance" {
```

```

+ disable_api_stop              = (known after apply)
+ disable_api_termination      = (known after apply)
+ ebs_optimized                 = (known after apply)
+ get_password_data            = false
+ host_id                      = (known after apply)
+ host_resource_group_arn      = (known after apply)
+ iam_instance_profile         = (known after apply)
+ id                           = (known after apply)
+ instance_initiated_shutdown_behavior = (known after apply)
+ instance_lifecycle            = (known after apply)
+ instance_state               = (known after apply)
+ instance_type                = "t2.micro"
+ ipv6_address_count           = (known after apply)
+ ipv6_addresses               = (known after apply)
+ key_name                     = (known after apply)
+ monitoring                   = (known after apply)
+ outpost_arn                  = (known after apply)
+ password_data                = (known after apply)
+ placement_group              = (known after apply)
+ placement_partition_number   = (known after apply)
+ primary_network_interface_id = (known after apply)
+ private_dns                  = (known after apply)
+ private_ip                   = (known after apply)
+ public_dns                   = (known after apply)
+ public_ip                    = (known after apply)
+ secondary_private_ips        = (known after apply)
+ security_groups              = (known after apply)
+ source_dest_check            = true
+ spot_instance_request_id     = (known after apply)
+ subnet_id                    = (known after apply)
+ tags                         = {
+   + "Name" = "UPES-EC2-Instnace"
+ }
+ tags_all                     = {
+   + "Name" = "UPES-EC2-Instnace"
+ }
+ tenancy                      = (known after apply)
+ user_data                    = (known after apply)
+ user_data_base64            = (known after apply)
+ user_data_replace_on_change = false
+ vpc_security_group_ids       = (known after apply)
}

Plan: 4 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_iam_user.iam_user[1]: Creating...
aws_iam_user.iam_user[2]: Creating...
aws_iam_user.iam_user[0]: Creating...
aws_instance.My-instance: Creating...
aws_iam_user.iam_user[1]: Creation complete after 2s [id=user2]
aws_iam_user.iam_user[2]: Creation complete after 2s [id=user3]
aws_iam_user.iam_user[0]: Creation complete after 2s [id=user1]
aws_instance.My-instance: Still creating... [10s elapsed]
aws_instance.My-instance: Still creating... [20s elapsed]
aws_instance.My-instance: Still creating... [30s elapsed]
aws_instance.My-instance: Creation complete after 33s [id=i-0d7b168226bb58756]

Apply complete! Resources: 4 added, 0 changed, 0 destroyed.

```

Instances (1) Info							
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>				Any state		< 1 > ⚙	
<input type="checkbox"/>	Name ↗	Instance ID	Instance state ▼	Instance type ▼	Status check	Alarm status	Availability Zone ▼
<input type="checkbox"/>	UPES-EC2-Inst...	i-0d7b168226bb58756	Running	t2.micro	⌚ Initializing	View alarms +	ap-south-1a

Step 7: Now perform Terraform destroy

```
F:\SEM 6\SPCM_LAB\SPCM_LAB_TERRAFORM>terraform destroy
aws_iam_user.iam_user[1]: Refreshing state... [id=user2]
aws_iam_user.iam_user[0]: Refreshing state... [id=user1]
aws_iam_user.iam_user[2]: Refreshing state... [id=user3]
aws_instance.My-instance: Refreshing state... [id=i-0d7b168226bb58756]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

- destroy

Terraform will perform the following actions:

```
# aws_iam_user.iam_user[0] will be destroyed
- resource "aws_iam_user" "iam_user" {
  - arn          = "arn:aws:iam::637423348062:user/user1" -> null
  - force_destroy = false -> null
  - id           = "user1" -> null
  - name         = "user1" -> null
  - path         = "/" -> null
  - tags         = {
    - "Name" = "user1-user"
  } -> null
  - tags_all     = {
    - "Name" = "user1-user"
  } -> null
  - unique_id    = "AIDAZI2LENFPPIYKH3ENU" -> null
}

# aws_iam_user.iam_user[1] will be destroyed
- resource "aws_iam_user" "iam_user" {
  - arn          = "arn:aws:iam::637423348062:user/user2" -> null
  - force_destroy = false -> null
  - id           = "user2" -> null
  - name         = "user2" -> null
  - path         = "/" -> null
  - tags         = {
    - "Name" = "user2-user"
  } -> null
  - tags_all     = {
    - "Name" = "user2-user"
  } -> null
  - unique_id    = "AIDAZI2LENFPA087OEGPA" -> null
}

# aws_iam_user.iam_user[2] will be destroyed
- resource "aws_iam_user" "iam_user" {
  - arn          = "arn:aws:iam::637423348062:user/user3" -> null
  - force_destroy = false -> null
  - id           = "user3" -> null
  - name         = "user3" -> null
  - path         = "/" -> null
  - tags         = {
    - "Name" = "user3-user"
  } -> null
  - tags_all     = {
    - "Name" = "user3-user"
  } -> null
  - unique_id    = "AIDAZI2LENFPEJ5KVHONZ" -> null
}

# aws_instance.My-instance will be destroyed
- resource "aws_instance" "My-instance" {
  - ami          = "ami-03f4878755434977f" -> null
  - arn          = "arn:aws:ec2:ap-south-1:637423348062:instance/i-0d7b168226bb58756" -> null
```

```

- get_password_data                = false -> null
- hibernation                      = false -> null
- id                              = "i-0d7b168226bb58756" -> null
- instance_initiated_shutdown_behavior = "stop" -> null
- instance_state                   = "running" -> null
- instance_type                   = "t2.micro" -> null
- ipv6_address_count               = 0 -> null
- ipv6_addresses                   = [] -> null
- monitoring                       = false -> null
- placement_partition_number       = 0 -> null
- primary_network_interface_id     = "eni-0a1385d40dcc7f88c" -> null
- private_dns                      = "ip-172-31-37-220.ap-south-1.compute.internal" -> null
- private_ip                      = "172.31.37.220" -> null
- public_dns                       = "ec2-43-205-230-157.ap-south-1.compute.amazonaws.com" -> null
- public_ip                       = "43.205.230.157" -> null
- secondary_private_ips            = [] -> null
- security_groups                  = [
  - "default",
] -> null
- source_dest_check                = true -> null
- subnet_id                       = "subnet-0fb95688eaa188f7d" -> null
- tags                             = {
  - "Name" = "UPES-EC2-Instnace"
} -> null
- tags_all                         = {
  - "Name" = "UPES-EC2-Instnace"
} -> null
- tenancy                         = "default" -> null
- user_data_replace_on_change     = false -> null
- vpc_security_group_ids          = [
  - "sg-0c6b5aae418c53ba2",
] -> null

- capacity_reservation_specification {
  - capacity_reservation_preference = "open" -> null
}

- cpu_options {
  - core_count      = 1 -> null
  - threads_per_core = 1 -> null
}

- credit_specification {
  - cpu_credits = "standard" -> null
}

- enclave_options {
  - enabled = false -> null
}

- maintenance_options {
  - auto_recovery = "default" -> null
}

- metadata_options {
  - http_endpoint      = "enabled" -> null
  - http_protocol_ipv6 = "disabled" -> null
  - http_put_response_hop_limit = 1 -> null
  - http_tokens        = "optional" -> null
  - instance_metadata_tags = "disabled" -> null
}

- private_dns_name_options {
  - enable_resource_name_dns_a_record = false -> null
}

```

```

    }

    - credit_specification {
      - cpu_credits = "standard" -> null
    }

    - enclave_options {
      - enabled = false -> null
    }

    - maintenance_options {
      - auto_recovery = "default" -> null
    }

    - metadata_options {
      - http_endpoint                = "enabled" -> null
      - http_protocol_ipv6           = "disabled" -> null
      - http_put_response_hop_limit = 1 -> null
      - http_tokens                   = "optional" -> null
      - instance_metadata_tags       = "disabled" -> null
    }

    - private_dns_name_options {
      - enable_resource_name_dns_a_record    = false -> null
      - enable_resource_name_dns_aaaa_record = false -> null
      - hostname_type                        = "ip-name" -> null
    }

    - root_block_device {
      - delete_on_termination = true -> null
      - device_name           = "/dev/sda1" -> null
      - encrypted              = false -> null
      - iops                   = 100 -> null
      - tags                   = {} -> null
      - throughput             = 0 -> null
      - volume_id              = "vol-05fdb35ff14989557" -> null
      - volume_size            = 8 -> null
      - volume_type            = "gp2" -> null
    }
  }
}

```

Plan: 0 to add, 0 to change, 4 to destroy.

Do you really want to destroy all resources?

Terraform will destroy all your managed infrastructure, as shown above.
There is no undo. Only 'yes' will be accepted to confirm.

Enter a value: yes

```

aws_iam_user.iam_user[0]: Destroying... [id=user1]
aws_iam_user.iam_user[1]: Destroying... [id=user2]
aws_iam_user.iam_user[2]: Destroying... [id=user3]
aws_instance.My-instance: Destroying... [id=i-0d7b168226bb58756]
aws_iam_user.iam_user[0]: Destruction complete after 2s
aws_iam_user.iam_user[1]: Destruction complete after 2s
aws_iam_user.iam_user[2]: Destruction complete after 2s
aws_instance.My-instance: Still destroying... [id=i-0d7b168226bb58756, 10s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0d7b168226bb58756, 20s elapsed]
aws_instance.My-instance: Still destroying... [id=i-0d7b168226bb58756, 30s elapsed]
aws_instance.My-instance: Destruction complete after 31s

```

Destroy complete! Resources: 4 destroyed.

Instances (1) Info							
<input type="text" value="Find Instance by attribute or tag (case-sensitive)"/>				Any state ▾		< 1 > ⚙	
<input type="checkbox"/>	Name ↗	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	Availability Zone ▾
<input type="checkbox"/>	UPES-EC2-Inst...	i-0d7b168226bb58756	Terminated 🔍	t2.micro	-	View alarms +	ap-south-1a

